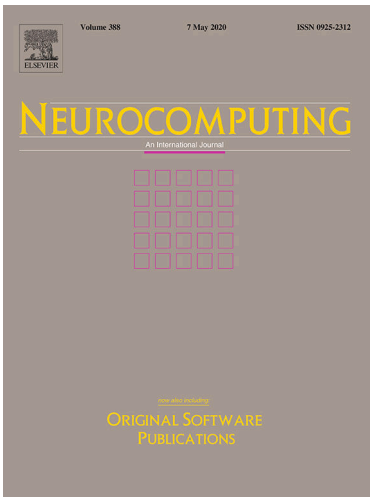
An memristor-based synapse implementation using BCM learning rule

Journal Pre-proofs



Brief papers

An memristor-based synapse implementation using BCM learning rule

Yongchuang Huang, Junxiu Liu, Jim Harkin, Liam McDaid, Yuling Luo

PII: S0925-2312(20)31727-6
DOI: https://doi.org/10.1016/j.neucom.2020.10.106
Reference: NEUCOM 23019

To appear in: *Neurocomputing*

Received Date: 15 February 2019
Accepted Date: 26 October 2020

Please cite this article as: Y. Huang, J. Liu, J. Harkin, L. McDaid, Y. Luo, An memristor-based synapse implementation using BCM learning rule, *Neurocomputing* (2020), doi: https://doi.org/10.1016/j.neucom.2020.10.106

An memristor-based synapse implementation using BCM learning rule

Yongchuang Huang[a], Junxiu Liu[a,*], Jim Harkin[b], Liam McDaid[b], Yuling Luo[a]

[a]*School of Electronic Engineering, Guangxi Normal University, Guilin, China*
[b]*School of Computing, Engineering and Intelligent Systems, Ulster University, Northern Ireland, United Kingdom, BT48 7JL*

Abstract

A novel memristive synapse model based on the HP memristor is proposed in this paper, which can address the problem of synaptic weight infinite modulations. The sliding threshold mechanism of the Bienenstock-Cooper-Munro rule (BCM) is used to redefine the memristance (i.e. synaptic weight) adjustment process of the memristive synapse model. Based on the proposed memristor-based synapse and Leaky Integrate-and-Fire neurons, a spiking neural network (SNN) hardware fragment is constructed, where spike trains with different frequencies are used to evaluate the stability performance of the proposed SNN hardware. Results show that compared to other approaches, the network is stable under different stimuli due to the characteristics of the memristor-based synapse model, and prove that the proposed synapse model is able to mimic biological synaptic behaviour and the problem of synaptic weight infinite modulations is addressed.

Keywords: memristor, BCM theory, spiking neural networks, learning rule

1. Introduction

Bio-inspired solutions have shown great potential for solving real-world engineering problems, and recent approaches have gained inspirations from biology to improve the reliability of electronic systems [1, 2, 3]. Various approaches

*Corresponding author
Email address: j.liu@ieee.com (Junxiu Liu)

have been proposed, e.g. using a plastic spiking neural network (SNN) model to develop a fault-resilient robotic controller [4], which has the potential to be applied in robotic obstacle avoidance task. Moreover, such fault-resilient system can maintain stability even under a synaptic fault density of up to 75%. In the approach of [5], a bio-inspired online fault detection and self-correction system for robotic spike-based controller is developed, where the self-detection and self-correction strategy can detect faults and re-allocate resources to restore the controller's functionality. Similarly, an astrocyte-neuron network for self-repairing mobile robotic car is proposed in the approach of [6], which can maintain the system stability through self-repair mechanism of the tripartite synapse. In the approach of [7], the Lyapunov-Krasovskii functional and linear matrix inequality framework are used to maintain the stability of the memristive recurrent neural networks.

In biology, the brain can adapt to external environment changes [8], where the learning mechanism plays a key role and the neural network can remain relatively stable (i.e. homeostatic [9]). Particularly, the Hebbian learning rule is a widely used neural network learning mechanism [10]. However, the conventional Hebbian learning rule has the drawback of unlimited modulation of synaptic weight, which can cause the system to collapse [10, 11]. Therefore, this learning rule was modified by introducing a sliding threshold, i.e. the Bienenstock-Cooper-Munro (BCM) learning rule, which can significantly improve the network stability [11, 12, 13]. Inspired by this, in our previous work [14], a self-repairing learning rule for spiking astrocyte-neuron networks is proposed. It is a combination of the spike-timing-dependent plasticity (STDP) and BCM learning rules, which can guarantee the system performance even with a synaptic fault density approaching 80%. Additionally, memristor is becoming popular in the biological neural networks. Due to that the characteristics of nonvolatility, nonlinearity and scalability, memristor device provides an elegant candidate for building the synapses and hardware neuromorphic systems, and various mathematical models of the memristor devices have been proposed and used for the synapse implementations [15, 16, 17, 18, 19]. However, due to the

physical properties of the memristor, i.e. the synaptic weight changes when it is stimulated under signals from the pre- or postsynaptic neurons, causing the memristor-based synapse faces the prominent problem of weight infinite modulation [11, 20, 21]. Considering that, if the synaptic weight of memristor-based synapse is modulated too high (i.e. memristance is very low), it leads to an overstimulated network where the postsynaptic neurons collapse [11]. Conversely, if the weight is too low (i.e. memristance is very high), the memristor-based synapse is in an inactive state [11]. Obviously, the synaptic weights affect the stability of the SNNs [22], but the traditional methods cannot well address the problem of synaptic weight infinite modulation. For instance, in order to avoid the synaptic weight modulation problem of the common oxide-based memristor, a binary memristive device of HfO_2-based oxide-based resistive memory (OxRAM) with only two distinct resistive states of low and high is applied in [20], where OxRAM cannot completely perform the behaviour of biological synapse in analogue neuromorphic circuit. In addition, some other methods make the weight modulation process of the memristive synaptic devices follow the unsupervised learning rules, e.g. STDP rule, through the specific external modification signal [21, 23]. Generating these external modified signals increases the complexity of the hardware implementation of memristive neuromorphic system. Therefore, different from these approaches, this paper aims to provide a more universal and simpler structure of SNN hardware for the memristive neuromorphic systems, and to solve the problem of synaptic weight infinite modulation of the common oxide-based memristive synapses. In summary, the main contributions of this work are as follows:

(a). A novel memristor-based synapse model is proposed in this paper which mimics the synaptic weight modulation process in the biological neural networks. Different from the conventional memristive synapse model, the proposed model uses the sliding threshold controlling mechanism to redefine the memristance adjustment process, which enables memristive synapses to be regulated by neuronal learning rules.

(b). The problem of synaptic weight infinite modulation in the conventional

memristive synapse is addressed by using the BCM learning rule. Based on the proposed memristor-based synapse model, a SNN hardware fragment is developed to show the scalability, where the change in network synaptic weight depends only on the activity level (i.e. spike rate) of the pre- and postsynaptic neurons.

(c). Results demonstrate that the proposed SNN architecture can maintain stability under the stimuli of continuous input spike trains with different frequencies, which is beneficial to build analogue hardware SNN circuits, and provide an elegant method to explore the reliable neuromorphic systems.

The paper is organized as follows. Section 2 gives the definitions of the learning rule and neuron model. Section 3 presents the memristor-based synapse model and the proposed SNN hardware architecture. Section 4 provides the experimental results under various input stimuli. Section 5 summarizes the paper and discusses the future works.

2. Learning Rule and LIF Neuron Model

This section analyses the constraints of Hebbian learning rule and gives a detailed definition of BCM learning rule and the LIF neuron model used in this paper. The Hebbian learning rule is one of the earliest synaptic plasticity learning rules, and it is described by:

$$\tau_w \frac{dw_{ij}}{\mathrm{d}t} = \upsilon_i u_j, \tag{1}$$

where ω_{ij} is the synaptic weight between neuron i and j, υ_i and u_j represent the spike rates of the pre- and postsynaptic neurons, respectively. τ_w is a time constant that controls the synaptic weight change rate. This conventional Hebbian learning rule is unstable as there is no decay for the synaptic weight to decrease [24]. When both pre- and postsynaptic neurons produce activity simultaneously, the synaptic weight keeps increasing, see the red solid in Fig. 1(a).

The BCM theory introduces the sliding threshold of θ_m based on Hebbian learning rule [12]. Its stability has been proved, namely metaplasticity [25, 26].

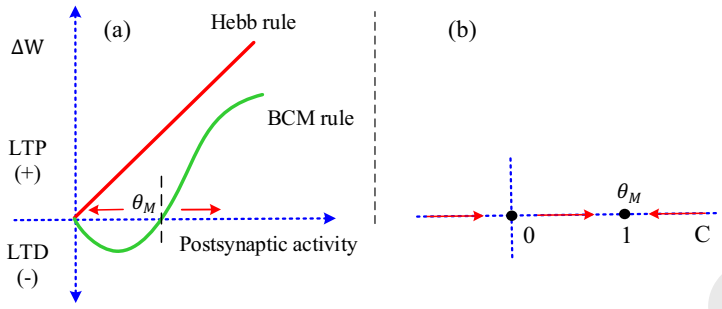


Figure 1: SNN learning rule. (a). BCM and Hebbian learning rules. (b). BCM instantaneous sliding threshold.

According to the BCM theory [27], the synaptic modification $\dot{m}(t)$ at time t is given by

$$\dot{m}(t) = \eta\phi(y(t), \theta_m(t))x(t) - \epsilon m(t), \tag{2}$$

where η is the constant learning rate, x and y represent the time-average of pre- and postsynaptic neurons activity, respectively. In this paper, the time-average of neuronal activity over the history is represented by the spike-count rate (i.e. mean frequency), and it can be calculated by $r = n/T$, where r denotes the spike-count rate, T is the running time and n is the number of spikes that appear during a trial. θ_m is the sliding modification threshold, ϵm is the decay term for synaptic efficacy. $\phi(\cdot)$ is the nonlinear activation for y and θ_m, and its definition on one-dimensional analysis is given by

$$\phi(y(t), \theta_m(t)) = y(t)(y(t) - \theta_m(t)). \tag{3}$$

In particular, the sliding threshold θ_m varies nonlinearly with the continuous time-weighted average output $\overline{\overline{y}}(t)$ over the history of the postsynaptic neurons. Hence the θ_m is also a frequency, and $\overline{\overline{y}}(t)$ is described by

$$\overline{\overline{y}}(t) = \frac{1}{\tau}\int_{-\infty}^{t} e^{-(t-t')/\tau}\, dt', \tag{4}$$

where τ is the exponential time window. A more stable form of θ_m was proposed later in the approach of [28], and described as

$$\theta_m(t) = \overline{\overline{y^p}} = \frac{1}{\tau}\int_{-\infty}^{t} y^p(t')e^{-(t-t')/\tau}\, dt', \tag{5}$$

where the degree of nonlinearity of θ_m is set by p, and $p = 2$ in this paper [26, 27], and θ_m is the squared activity integrated over the exponential time window τ which is given by

$$\tau = \int_{-\infty}^{t} e^{-(t-t')/\tau}\, dt'. \tag{6}$$

It should be noted that because of the initial dormancy of the synaptic rule links, $y = 0$ if $t < 0$ [26]. Then eqn. (5) can be rewritten as

$$\theta_m(t) = \frac{\int_0^t y^2(t') e^{-(t-t')/\tau}\, dt'}{\int_0^t e^{-(t-t')/\tau}\, dt'}, \tag{7}$$

where the change speed of θ_m affects the activity of postsynaptic neurons. Particularly, if θ_m changes too slowly (i.e. τ is too large), it will cause large oscillations of postsynaptic neuron activity. θ_m will become large with rapid fluctuations if it is updated instantaneously (i.e. τ is too small). Obviously, the value of τ has a directly effect on the network stability. In order to improve the network stability, the sensitivity analysis of the parameters η, τ and x is necessary [29, 30]. Thus, according to the previous work of [27], for a given learning rate η, the sensitivity description between these parameters of η, τ and x can be described as $\eta\tau x<1$, where η is experimentally set to $\eta = 10^{-5}$ in this paper.

As shown in Fig. 1(a), the sliding threshold θ_m oscillates with the level of postsynaptic activity, according to the BCM theory (see the green solid). When the postsynaptic activity level is lower than θ_m, a long-term depression (LTD) effect occurs on synaptic weights ($\Delta W < 0$). If it is greater than θ_m, a long-term potentiation (LTP) effect happens, i.e. $\Delta W > 0$. Due to the sliding threshold and gentle BCM curve, the infinite modulation of the synaptic weight is prevented and the postsynaptic neurons are finally in a dynamic stabile state. In combination with the bipolar characteristic of the memristor-based synapse, for the LTD the memristor-based synapse outputs a negative electrical signal, which is contrary to the BCM theory. Therefore, the instantaneous sliding threshold [27] is used in this paper to study the stability of the memristor-based SNNs.

One significant feature of the instantaneous sliding threshold is $C = mx(t)$, where C and x denote the activity levels of pre- and postsynaptic neurons, respectively, m denotes the current synaptic weight. Note that C is different from the time-average activity $y(t)$ of the postsynaptic neuron in (3). The calculation of θ_m is unchanged and the $y(t)$ in (3) is replaced by C, so eqn. (3) can be rewritten as

$$\phi(C, \theta_m(t)) = C(C - \theta_m(t)). \tag{8}$$

Fig. 1(b) shows the only stable point in this case is $\theta_m = C = 1$. When $\theta_m = 1$, the neural network is dynamically stable. This is the key to the instantaneous sliding threshold combined with the memristor-based synapse, and the detailed mathematical analysis will be derived in section 3.

Additionally, the LIF neuron model is used in this work due to its simplistic nature, which can be described by

$$\tau_m \frac{\mathrm{d}v}{\mathrm{d}t} = -v(t) + R_m \sum_{k=1}^{n} I_{syn}^{k}(t), \tag{9}$$

where τ_m is the neuronal membrane time constant, v is the membrane potential, R_m is the membrane resistance, $I_{syn}^{k}(t)$ denotes the current injected by k^{th} synapse, and n represents the total number of synapses connected to the neuron. The firing threshold voltage is 9 mv.

3. Memristor-based Synapse and MSNN Architecture

In this section, the characteristics of the memristor-based synapse model and the SNN hardware architecture with two neurons are provided. The mathematical definition and characteristic simulations of the memristor are given firstly; then the combination between memristor and BCM rule is analysed. Finally, the proposed two-neuron network architecture is presented in detail.

3.1. Memristor model and its combination with BCM rule

The TiO_2-based memristor is used as the synapse model in this paper due to its nonvolatility, nonlinearity and dynamic properties [18]. This type of memristor model was first proposed by HP Labs in 2008 [31], and its corresponding

physical model and symbol are shown in Fig. 2(a) and (b) respectively. It is a two-end component with positive and negative ("+" and "-"), comprising a doped region with TiO_{2-x} (w) and an undoped region with TiO_2 ($D-w$), respectively. The doped region has high conductivity, while the undoped region has high insulation, and the total thickness of the two regions is represented by D as a fixed value of 10 nm. This memristor model is bipolar. If the input is positive current, the thickness of the doped region increases, and the resistance of the memristor decreases. If the input is negative current, the thickness of the undoped region increases and the resistance increases.

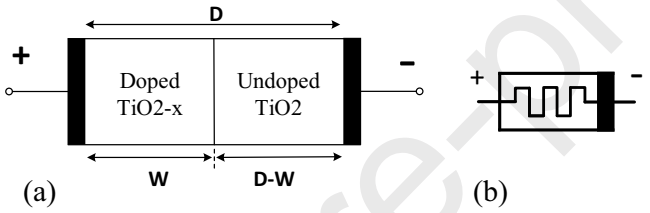


Figure 2: The TiO_2-based memristor [31]. (a). Physical model. (b). Memristor symbol.

The TiO_2-based memristor model is given by

$$R_{mem}(X(t)) = R_{off} - X(t)\Delta R, \tag{10}$$

where R_{mem} is the total resistance of the memristor which equals to the sum of the resistance of the doped and undoped regions, R_{off} and R_{on} denote the maximum and minimum resistances corresponding to the case of $w = 0$ and $w = D$, $\Delta R = R_{off} - R_{on}$, and $X(t) = w(t)/D$. $X(t)$ represents the ratio between the doped region and the total thickness of the oxide layer. It is always used as the weight of the memristor-based synapse [32], and $X \in (0, 1)$. This TiO_2-based memristor is a charge-controlled model and its voltage can be described by

$$v(t) = R_{mem}(X(t))i(t), \tag{11}$$

where $v(t)$ and $i(t)$ are the voltage and current, respectively, and the change rate of $X(t)$ is given by

$$\mathrm{d}X/\mathrm{d}t = ki(t)f(X), \tag{12}$$

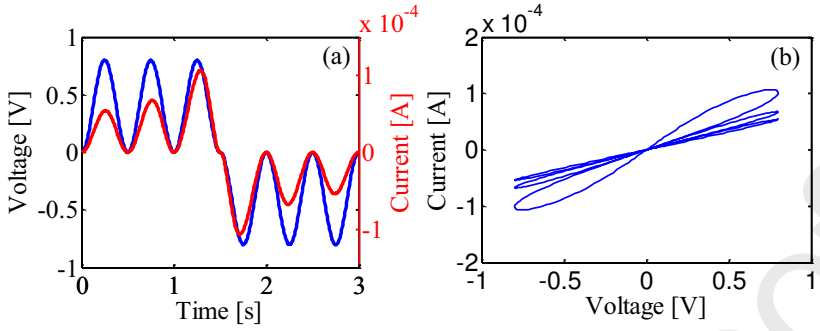


Figure 3: Simulation results of memristor characteristics. The corresponding parameters of the memristor are: $R_{off} = 20k$, $R_{on} = 100$, $R_init = 10k$, $p = 2$, $D = 10nm$. (a). The memristor current varies with the input voltage. (b). The hysteresis curve of the memristor.

where $k = \mu_v R_{on}/D^2$, and $\mu_v = 10^{-14} m^2 s^{-1} v^{-1}$ is the average ion mobility, R_{on} is the minimum memristor resistance, and $f(X)$ is the Joglekar's window function [33] which can be described by

$$f(X) = 1 - (2X(t) - 1)^{2p}, \tag{13}$$

where p controls the degree of nonlinearity, and $p = 2$ in this paper.

Fig. 3 shows the simulation results of the memristor characteristics with an input of $v(t) = \pm 0.8 \sin^2(2\pi t)$. As can be seen by Fig. 3(a), when the input is a positive voltage, the current changes with the voltage and gradually increases, indicating that the memristor resistance is decreasing, and vice versa. Fig. 3(b) shows one of the most significant features of the memristor with this input, i.e. the hysteresis characteristics.

Memristive devices have been used for the SNN synapse implementation based on a specific input signal [11, 23]. However, in addition to synapses, neurons are another key component of biological neural networks. Therefore, a SNN fragment of two neurons and its hardware architecture is proposed in this work, where the BCM is used as the learning rule. Specifically, the instantaneous sliding threshold of the BCM rule is employed to supervise the change process

of the weight of the memristor-based synapse, so that eqn. (12) is rewritten as

$$\frac{\mathrm{d}X}{\mathrm{d}t} = \begin{cases} ki(t)f(X)\theta(|\dot{\theta}_m|), & \text{if } |\dot{\theta}_m| \neq 0 \\ 0, & \text{if } |\dot{\theta}_m| = 0 \end{cases}, \tag{14}$$

where $\theta(\cdot)$ is a step function, $|\dot{\theta}_m|$ is the absolute value of the sliding threshold increment. In particular, $|\dot{\theta}_m|$ is updated at every iteration, i.e θ_m changes with the activity level of the postsynaptic neuron. Obviously, through eqn. (14), an inherent relationship between the sliding threshold θ_m, the weight of memristor-based synapse and the network output is set up.

According to the BCM theory, the sliding threshold changes quickly enough to catch up the neuron activities, and the postsynaptic neurons keep the output by adjusting the synaptic weight [27]. When the output approaches stable state, the instantaneous sliding threshold θ_m converges to a constant value (i.e. one), as shown in Fig. 1(b). This property of θ_m contributes to the stability of the proposed SNN architecture. As can be seen from eqn. (14), when $|\dot{\theta}_m| \neq 0$, the output is not converged. At this time, $\theta(|\dot{\theta}_m|) = 1$, then eqn. (14) is equivalent to eqn. (12), i.e. the convergence process of the memristor-based synapse is not affected. When $|\dot{\theta}_m| = 0$, the neural network is dynamically stable, and the change rate of the memristor-based synaptic weight is 0, then the output remains constant. Obviously, by combining the BCM sliding threshold with the memristor model, the supervision of the synaptic weight change process is realized.

In addition, the current synaptic efficacy $m(t)$ in eqn. (2) is considered as the input of the memristor-based synapse at time t, which is used to analyse the regulation of the BCM mechanism on the memristive synapse. The current $i(t)$ of (14) can be calculated by

$$i(t) = \frac{m(t)}{R_{off} - X(t)\Delta R}, \tag{15}$$

where $\Delta R = R_{off} - R_{on}$. From eqn. (2), (14) and (15), it can be clearly seen that the weight change of the memristor-based synapse depends on the activity levels of the pre- and postsynaptic neurons, which is biologically plausible [34].

At the same time, the change of θ_m has a direct impact on the passing current $i(t)$ of the memristor-based synapse which is used to modulate the synaptic weights (i.e. memristance). By adjusting the synaptic weights, the infinite modulation of the synapse is prevented, and the stability of the network is achieved.

3.2. Memristor-based SNN architecture

Fig. 4 illustrates the structure of the proposed SNN fragment. It contains two LIF neurons (i.e. input neuron $N1$ and output neuron $N2$) and a memristor-based synapse (excitatory). When the presynaptic neuron $N1$ is excited by the injected current I_{in}, it outputs a spike train. The rate of the spike train is the input to the synapse, which is denoted by x in (2). Due to the high initial weight, the output current I_m of the synapse can induce activity of the postsynaptic neuron $N2$. Then the activity level of $N2$, i.e. the rate of the output spike train y, is fed back to the memristor-based synapse, and the change rate of synaptic weight is adjusted by BCM learning rule. Because the sliding modification threshold θ_m moves quickly enough to catch up with the activity of the neurons, and it eventually converges to a certain value (i.e. one). At this time, the synaptic weight change rate of memristor-based synapse is 0, as shown in eqn. (14). This causes the output spike frequency of $N2$ to converge, indicating that the system is dynamically stable. By combining the BCM rule with the memristor model, the problem of synaptic infinite modulation under the continuous stimulus is solved, which provides an alternative method to improve

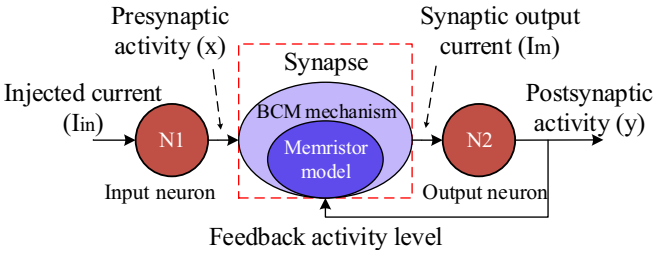


Figure 4: A SNN fragment.

the stability of SNNs. This structure has the potential to be scaled to large networks, i.e. multiple synapses and neurons.

4. Results

This section provides the results of the proposed SNN architecture where the stability is analysed under input spike trains with various frequencies. The Matlab simulation platform is used for all simulations and the memristor modelling. The time step of simulation is set as a fixed value of 1ms which is same as the time step of Euler method of integration. The corresponding parameters of the memristor-based synapse are $R_{off} = 10k$, $R_{on} = 100$, $R_{init} = 5k$, $p = 2$, $D = 10nm$, i.e. the initial synaptic weight value is 0.5. According to the BCM instantaneous sliding threshold, the synaptic weight increases in proportion to $1/x$ [27], where x is the spike frequency of the presynaptic neuron. A spike train with low frequency is more likely to enhance synaptic efficacy, which is also detailed in [11]. Based on this, the stability of the network is firstly analysed, where the memristor model does not incorporate BCM learning rule. In this case, the memristor-based synaptic weight is changed according to (12), i.e. the BCM rule cannot modulate the synaptic weight change process. The corresponding simulation results are shown in Fig. 5(a)-(d). The spike-count rate is considered as the activity levels of neuron $N1$, $N2$, see Fig. 4. When $N1$ has a continuous stimulus and produces a spike train of $\sim$10Hz, the sliding threshold θ_m can still self-adjust and eventually converge to one, see Fig. 5(a) and (b). However, the synaptic weight is not modulated by the BCM, and it grows indefinitely with the continuous stimulus $x(t)$, and eventually exceeds the variation range ($X \in (0, 1)$) of the memristor-based synapse, i.e. $X = 1$ (Fig. 5(c)). The infinite modulation of the synapse causes the $N2$ output spike train frequency to increase continuously indicating that the system is unstable, see Fig. 5(d). In order to prevent the infinite modulation of the synapse, the BCM is adopted to modulate the synaptic weight by combining θ_m with the memristor model in (14). The experiment is repeated again. As shown by Fig. 5(f), the output

$y(t)$ of $N2$ tends to stabilize at ∼345Hz as the sliding threshold θ_m converges. Meanwhile, compared to the previous result in Fig. 5(c), the synaptic weight is maintained at 0.59, see Fig. 5(e).

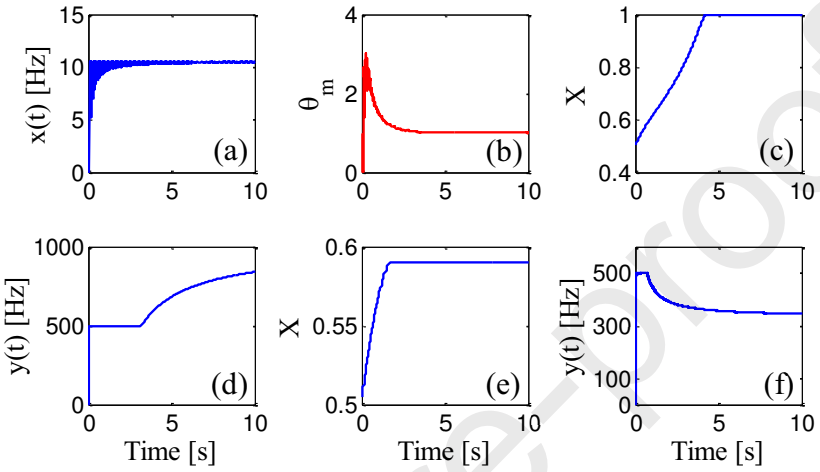


Figure 5: System stability simulation results when the output spike frequency of presynaptic neuron is ∼10Hz.

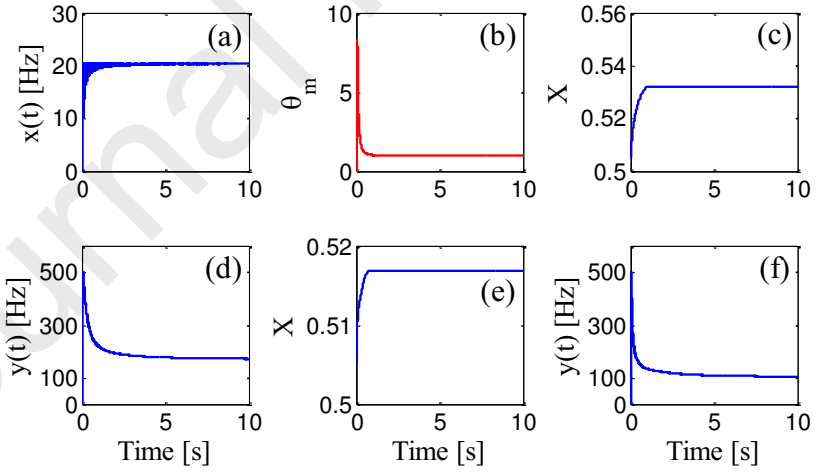


Figure 6: System stability simulation results when the output spike frequency of presynaptic neuron is ∼20Hz (a-d) and ∼30Hz (e, f), respectively.

To further evaluate the stability performance of the proposed architecture, it

is verified by using different input spike trains of ∼20Hz and ∼30Hz respectively. Fig. 6(a)-(d) show the results under the N1 output spike train of ∼20Hz. Similar to the previous discussions, both the synaptic weight X and the output activity level $y(t)$ are gradually stabilized at 0.53 and ∼172Hz, respectively, as the sliding threshold θ_m converges. A similar profile can also be observed when the input spike train is ∼30Hz as presented in Fig. 6(e) and (f), where the corresponding final weight value X and $y(t)$ are 0.52 and ∼104Hz, respectively. Based on the results in Fig. 5 and Fig. 6, it can be seen that the greater the activity intensity of the presynaptic neuron $N1$, the smaller the final synaptic weight value. This is consistent with the biological experimental results in the approach of [27]. In addition, the proposed structure contains the neuron and synapse components, which is more complete and adaptable compared to other approaches, e.g. the approaches of [11, 23] are based on one device to mimic biological behaviours of synapse.

In this paper, we aim to address the synaptic weight infinite modulation problem of the common oxide-based analogue memristive synapse, and provide a stable SNN framework for neural network hardware implementations. table I shows the comparison between different methods for neural network implementations on three different aspects including synapse model, network architecture and learning rule/synthesis methodology (LR/SM). The proposed SNN is constructed by using the analogue memristive synapse model, and can perform the synaptic behaviours of biological synapse, while the approach of [20] uses multiple parallel binary HfO_2-based OxRAM cells to model the biological synaptic behaviour at the expense of increased area consumption [21]. In addition, the regulation signal of the proposed memristor-based synapse model is spike trains, while the device programming in the approaches of [21, 23] requires extra circuit elements for monitoring the state of the memristor and shaping the spike accordingly, which consume more area and power. Other approaches of [35, 36, 37] use hybrid algorithm, STEERAGE and SCANN synthesis methodology, to synthesize efficient artificial neural networks for forecasting [35] and pattern recognition [36, 37] tasks. Compared to these approaches [35, 36, 37],

the memristive neuromorphic networks show advantages for hardware implementations [17, 21, 23], especially the high system integration as the memristor is nanodevice with very small area overhead and power consumption [17, 18]. Particularly, due to the fast information processing capability of the hardware circuit system (especially the reaction time of the memristor device can reach the nanosecond level [17]), the memristive SNN hardware circuit has the potential of processing information at the biological time scale.

Table I: Comparison between different methods for neural network implementations

Approaches	Synapse model	Network architecture	LR/SM
[20]	OxRAM	CNN	STDP
[21]	HfO_2-based	SNN	STDP
[23]	Iron oxide-based	–	STDP & BCM
[35]	–	Hybrid networks	Hybrid algorithm
[36]	–	ANN	SCANN
[37]	–	FFNN & CNN	STEERAGE
This work	TiO_2-based	SNN	BCM

5. Conclusion

An optimized memristor-based synapse model is proposed in this work. By using the proposed synapse and LIF neuron models, a SNN hardware is developed which employs the BCM mechanism as learning rule. The stability of the proposed SNN hardware is evaluated under the input spike trains with different frequencies from 10Hz to 30Hz. Results demonstrate that the SNN has a stable network output and can maintain stability under a continuous excitatory stimulus (represented by spike trains). The proposed memristor-based synapse model is able to mimic biological synaptic behaviour and address the problem of synaptic weight infinite modulations. It has the potential to be scaled to large neural networks due to its compact structure. Future works will explore

the stability of networks on the architecture level and the analogue hardware architecture modelling.

Acknowledgements

This research is supported by the National Natural Science Foundation of China under Grant 61976063 and the funding of Overseas 100 Talents Program of Guangxi Higher Education.

References

[1] J. Liu, L. J. McDaid, J. Harkin, S. Karim, A. P. Johnson, A. G. Millard, J. Hilder, D. M. Halliday, A. M. Tyrrell, J. Timmis, Exploring self-repair in a coupled spiking astrocyte neural network, IEEE Transactions on Neural Networks and Learning Systems In Press (2018) 1–11.

[2] Y. Luo, L. Wan, J. Liu, J. Harkin, L. McDaid, Y. Cao, X. Ding, Low cost interconnected architecture for the hardware spiking neural networks, Frontiers in Neuroscience 12 (1) (2018) 1–14.

[3] J. Liu, J. Harkin, L. P. Maguire, L. J. Mcdaid, J. J. Wade, SPANNER: A self - repairing spiking neural network hardware architecture, IEEE Transactions on Neural Networks and Learning Systems 29 (4) (2018) 1287–1300.

[4] A. P. Johnson, J. Liu, A. G. Millard, S. Karim, A. M. Tyrrell, J. Harkin, J. Timmis, L. J. McDaid, D. M. Halliday, Homeostatic fault tolerance in spiking neural networks: A dynamic hardware perspective, IEEE Transactions on Circuits and Systems I: Regular Papers 65 (2) (2017) 687–699.

[5] J. Liu, J. Harkin, M. Mcelholm, L. Mcdaid, Case study : Bio-inspired self-adaptive strategy for spike-based PID controller, in: IEEE International Symposium on Circuits and Systems (ISCAS), 2015, pp. 2700–2703.

[6] J. Liu, J. Harkin, L. Maguire, L. McDaid, J. Wade, M. McElholm, Self-repairing hardware with astrocyte-neuron networks, in: IEEE International Symposium on Circuits and Systems (ISCAS), 2016, pp. 1350–1353.

[7] K. Mathiyalagan, R. Anbuvithya, R. Sakthivel, J. H. Park, P. Prakash, Reliable stabilization for memristor-based recurrent neural networks with time-varying delays, Neurocomputing 153 (1) (2015) 140–147.

[8] M. Opendak, E. Gould, Adult neurogenesis: A substrate for experience-dependent change, Trends in Cognitive Sciences 19 (3) (2015) 151–161.

[9] G. W. Davis, I. Bezprozvanny, Maintaining the stability of neural function: A homeostatic hypothesis, Annual review of physiology 63 (1) (2001) 847–869.

[10] D. O. Hebb, The organization of behavior: A neuropsychological theory, New York: Wiley, 1949.

[11] W. Dong, F. Zeng, Y. Hu, C. Chang, X. Li, F. Pan, G. Li, Sliding threshold of spike-rate dependent plasticity of a semiconducting polymer/electrolyte cell, Journal of Polymer Science, Part B: Polymer Physics 54 (23) (2016) 2412–2417.

[12] E. L. Bienenstock, L. N. Cooper, P. W. Munro, Theory for the development of neuron selectivity: orientation specificity and binocular interaction in visual cortex, The Journal of Neuronscience 2 (1) (1982) 32–48.

[13] L. N. Cooper, M. F. Bear, The BCM theory of synapse modification at 30:interaction of theory with experiment, Nature Reviews Neuroscience 13 (11) (2012) 798–810.

[14] J. Liu, L. J. Mcdaid, J. Harkin, J. J. Wade, S. Karim, A. P. Johnson, A. G. Millard, D. M. Halliday, A. M. Tyrrell, J. Timmis, Self-repairing learning rule for spiking astrocyte-neuron networks, in: International Conference on Neural Information Processing (ICONIP), 2017, pp. 384–392.

[15] C. Yakopcic, T. M. Taha, G. Subramanyam, R. E. Pino, S. Rogers, A memristor device model, IEEE Electron Device Letters 32 (10) (2011) 1436–1438.

[16] I. Vourkas, A. Batsos, G. C. Sirakoulis, SPICE modeling of nonlinear memristive behavior, International Journal of Circuit Theory and Applications 43 (5) (2015) 553–565.

[17] J. Liu, Y. Huang, Y. Luo, J. Harkin, L. McDaid, Bio-inspired fault detection circuits based on synapse and spiking neuron models, Neurocomputing 331 (1) (2018) 473–482.

[18] X. Liu, Z. Zeng, S. Wen, Implementation of memristive neural network with full-function pavlov associative memory, IEEE Transactions on Circuits and Systems I: Regular Papers 63 (9) (2016) 1454–1463.

[19] K. D. Cantley, A. Subramaniam, H. J. Stiegler, R. A. Chapman, E. M. Vogel, Hebbian learning in spiking neural networks with nanocrystalline silicon TFTs and memristive synapses, IEEE Transactions on Nanotechnology 10 (5) (Sep. 2011) 1066–1073.

[20] D. Garbin, E. Vianello, O. Bichler, Q. Rafhay, C. Gamrat, G. Ghibaudo, B. DeSalvo, L. Perniola, HfO_2-based OxRAM devices as synapses for convolutional neural networks, IEEE Transactions on Electron Devices 62 (8) (2015) 2494–2501.

[21] E. Covi, S. Brivio, A. Serb, T. Prodromakis, M. Fanciulli, S. Spiga, Analog memristive synapse in spiking networks implementing unsupervised learning, Frontiers in Neuroscience 10 (1) (2016) 1–13.

[22] Y. Luo, Q. Fu, J. Liu, Y. Huang, X. Ding, Y. Cao, Improving the stability for spiking neural networks using anti-noise learning rule, in: Pacific Rim International Conference on Artificial Intelligence (PRICAI), 2018, pp. 29–37.

[23] W. He, K. Huang, N. Ning, K. Ramanathan, G. Li, Y. Jiang, J. Y. Sze, L. Shi, R. Zhao, J. Pei, Enabling an integrated rate-temporal learning scheme on memristor, Scientific Reports 4 (1) (2014) 1–6.

[24] F. Zenke, W. Gerstner, S. Ganguli, The temporal paradox of Hebbian learning and homeostatic plasticity, Current Opinion in Neurobiology 43 (1) (2017) 166–176.

[25] W. C. Abraham, M. F. Bear, Metaplasticity: The plasticity of synaptic plasticity, Trends in Neurosciences 19 (4) (1996) 126–130.

[26] J. Tan, C. Quek, A BCM theory of meta-plasticity for online self-reorganizing fuzzy-associative learning, IEEE Transactions on Neural Networks 21 (6) (Jun. 2010) 985–1003.

[27] L. N. Cooper, N. Intrator, B. S. Blais, H. Z. Shouval, Theory of Cortical Plasticity, Singapore: World Scientific, 2004.

[28] N. Intrator, L. N. Cooper, Objective function formulation of the BCM theory of visual cortical plasticity : statistical connections , stability conditions, Neural Networks 5 (1) (1992) 3–17.

[29] A. Sharifi, M. Ahmadi, H. Badfar, M. Hosseini, Modeling and sensitivity analysis of NOx emissions and mechanical efficiency for diesel engine, Environmental Science and Pollution Research 26 (24) (2019) 25190–25207.

[30] S. Dorosti, S. Jafarzadeh Ghoushchi, E. Sobhrakhshankhah, M. Ahmadi, A. Sharifi, Application of gene expression programming and sensitivity analyses in analyzing effective parameters in gastric cancer tumor size and location, Soft Computing 24 (13) (2020) 9943–9964.

[31] D. B. Strukov, G. S. Snider, D. R. Stewart, R. S. Williams, The missing memristor found, Nature 534 (7191) (2008) 80–83.

[32] Y. Zhang, Z. Zeng, S. Wen, Implementation of memristive neural networks with spike-rate-dependent plasticity synapses, in: International Joint Conference on Neural Networks (IJCNN), 2014, pp. 2226–2233.

[33] Y. N. Joglekar, S. J. Wolf, The elusive memristor: Properties of basic electrical circuits, European Journal of Physics 30 (4) (2009) 661–675.

[34] C. D. Schuman, T. E. Potok, R. M. Patton, D. Birdwell, M. E. Dean, G. S. Rose, S. James, A survey of neuromorphic computing and neural networks in hardware, Neural and Evolutionary Computing 1705 (06963) (2017) 1–88.

[35] M. Ahmadi, S. Jafarzadeh Ghoushchi, R. Taghizadeh, A. Sharifi, Presentation of a new hybrid approach for forecasting economic growth using artificial intelligence approaches, Neural Computing and Applications 31 (12) (2019) 8661–8680.

[36] S. Hassantabar, Z. Wang, N. K. Jha, SCANN: synthesis of compact and accurate neural networks, arXiv preprint arXiv:1904.09090 (2019) 1–11.

[37] S. Hassantabar, X. Dai, N. K. Jha, STEERAGE: synthesis of neural networks using architecture search and grow-and-prune methods, arXiv preprint arXiv:1912.05831 (2019) 1–12.